

# Semi-Implicit Update

## 1 Derivation

At this point in the simulation, we want to update the grid node velocity. Explicitly, this is just

$$\mathbf{v}_i^{n+1} = \mathbf{v}_i^n + \frac{\partial \mathbf{v}_i^n}{\partial n} = \mathbf{v}_i^n + \Delta t \frac{\partial \mathbf{v}_i^n}{\partial t} = \mathbf{v}_i^n + \Delta t \frac{\mathbf{f}_i^n}{m_i}, \quad (1)$$

by Newton's second law. The backward Euler method can be used to express the new velocity implicitly:

$$\mathbf{v}_i^{n+1} = \mathbf{v}_i^n + \frac{\partial \mathbf{v}_i^{n+1}}{\partial n} = \mathbf{v}_i^n + \Delta t \frac{\partial \mathbf{v}_i^{n+1}}{\partial t} = \mathbf{v}_i^n + \Delta t \frac{\mathbf{f}_i^{n+1}}{m_i}. \quad (2)$$

The “semi-implicit” update employed by the authors essentially interpolates between these two schemes:

$$\begin{aligned} \mathbf{v}_i^{n+1} &= \mathbf{v}_i^n + (1 - \beta) \Delta t \frac{\mathbf{f}_i^n}{m_i} + \beta \Delta t \frac{\mathbf{f}_i^{n+1}}{m_i} \\ &= \mathbf{v}_i^n + \frac{\Delta t}{m_i} \left( \mathbf{f}_i^n + \beta (\mathbf{f}_i^{n+1} - \mathbf{f}_i^n) \right) \\ &\approx \mathbf{v}_i^n + \frac{\Delta t}{m_i} \left( \mathbf{f}_i^n + \beta \Delta t \frac{\partial \mathbf{f}_i^n}{\partial t} \right). \end{aligned}$$

Note that

$$\frac{\partial \mathbf{f}_i^n}{\partial t} \approx \sum_j \frac{\partial \mathbf{f}_i^n}{\partial \hat{\mathbf{x}}_j} \mathbf{v}_j^{n+1},$$

where  $\hat{\mathbf{x}}_i = \mathbf{x}_i + \Delta t \mathbf{v}_i$ . So, the update scheme is

$$\mathbf{v}_i^{n+1} = \mathbf{v}_i^n + \frac{\Delta t}{m_i} \left( \mathbf{f}_i^n + \beta \Delta t \sum_j \frac{\partial \mathbf{f}_i^n}{\partial \hat{\mathbf{x}}_j} \mathbf{v}_j^{n+1} \right). \quad (3)$$

You can see that when  $\beta = 0$ , this update is equivalent to the explicit update, and when  $\beta = 1$ , this update is (approximately) equivalent to the backward Euler update.

Define the explicitly updated velocity as

$$\mathbf{v}_i^* = \mathbf{v}_i^n + \Delta t \frac{\mathbf{f}_i^n}{m_i} \quad (4)$$

By definition,

$$\frac{\partial^2 \Phi^n}{\partial \hat{\mathbf{x}}_i \partial \hat{\mathbf{x}}_j} = - \frac{\partial \mathbf{f}_i^n}{\partial \hat{\mathbf{x}}_j}. \quad (5)$$

Using (3), (4), and (5), we have that

$$\mathbf{v}_i^* = \mathbf{v}_i^{n+1} + \frac{\beta \Delta t^2}{m_i} \sum_j \frac{\partial^2 \Phi^n}{\partial \hat{\mathbf{x}}_i \partial \hat{\mathbf{x}}_j} \mathbf{v}_j^{n+1} = \sum_j \left( I \delta_{ij} + \frac{\beta \Delta t^2}{m_i} \frac{\partial^2 \Phi^n}{\partial \hat{\mathbf{x}}_i \partial \hat{\mathbf{x}}_j} \right) \mathbf{v}_j^{n+1}. \quad (6)$$

## 2 Implementation

This update can be formulated as a large system of linear equations. Define  $\mathbf{V}$  as the vector of all the updated grid node velocities, such that

$$\mathbf{v} \triangleq \left( v_{1x}^{n+1}, v_{1y}^{n+1}, v_{1z}^{n+1}, \dots, v_{gx}^{n+1}, v_{gy}^{n+1}, v_{gz}^{n+1} \right)^T,$$

where  $g$  is the total number of grid nodes. Likewise, define  $\mathbf{V}^*$  as the vector of all the explicitly update grid node velocities, such that

$$\mathbf{v}^* \triangleq \left( v_{1x}^*, v_{1y}^*, v_{1z}^*, \dots, v_{gx}^*, v_{gy}^*, v_{gz}^* \right)^T.$$

We want to express (6) in matrix form as  $\mathbf{v}^* = \mathbf{E}\mathbf{v}$ . Both  $\mathbf{v}$  and  $\mathbf{v}^*$  have dimensions  $3g \times 1$ , so  $\mathbf{E}$  is a matrix of dimensions  $3g \times 3g$ . The conjugate residual method is used to solve this large linear system.

### 2.1 Conjugate Residual Method

The conjugate residual method is an iterative method used to solve linear systems of the general form  $\mathbf{A}\mathbf{x} = \mathbf{b}$ . We'll use  $\mathbf{v}^*$  as our initial guess  $\mathbf{v}_0$ . The algorithm proceeds as follows:

```

 $\mathbf{r}_0 = \mathbf{v}^* - \mathbf{E}\mathbf{v}_0$ 
 $\mathbf{s}_0 = \mathbf{E}\mathbf{r}_0$ 
 $\mathbf{p}_0 = \mathbf{r}_0$ 
 $\mathbf{q}_0 = \mathbf{s}_0$ 
 $\gamma_0 = \mathbf{r}_0^T \mathbf{s}_0$ 
for  $k = 0, 1, 2, \dots$  do
     $\alpha_k = \frac{\gamma_k}{\mathbf{q}_k^T \mathbf{q}_k}$ 
     $\mathbf{v}_{k+1} = \mathbf{v}_k + \alpha_k \mathbf{p}_k$ 
     $\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{q}_k$ 
     $\mathbf{s}_{k+1} = \mathbf{E}\mathbf{r}_{k+1}$ 
     $\beta_k = \frac{\mathbf{r}_{k+1}^T \mathbf{s}_{k+1}}{\gamma_k}$ 
     $\gamma_{k+1} = \beta_k \gamma_k$ 
     $\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k$ 
     $\mathbf{q}_{k+1} = \mathbf{s}_{k+1} + \beta_k \mathbf{q}_k$ 
end for

```

This will continue until some stopping criterion is satisfied; presumably, when  $\|\alpha_k \mathbf{p}_k\|^2$  falls below some threshold. The important thing to note is that we don't actually have to explicitly define the matrix  $\mathbf{E}$ , we just need to be able to compute  $\mathbf{E}\mathbf{u}$  given some vector  $\mathbf{u}$ .

### 2.2 Computation of $\mathbf{E}\mathbf{u}$

The key term in (6) is the Hessian of the potential. The "action" of this Hessian on an arbitrary increment  $\delta\mathbf{u}$  is

$$\delta\mathbf{f}_i = - \sum_j \frac{\partial^2 \Phi}{\partial \hat{\mathbf{x}}_i \partial \hat{\mathbf{x}}_j} \delta\mathbf{u}_j = - \sum_p V_p^0 \mathbf{A}_p (\mathbf{F}_{Ep}^n)^T \nabla w_{jp}^n, \quad (7)$$

where

$$\mathbf{A}_p = \frac{\partial^2 \Psi}{\partial \mathbf{F}_{Ep} \partial \mathbf{F}_{Ep}} : \left( \sum_j \delta\mathbf{u}_j (\nabla w_{jp})^T \mathbf{F}_{Ep}^n \right). \quad (8)$$

Generally, the notation  $a = \mathbf{B} : \mathbf{C}$  indicates the Frobenius inner product of matrices  $\mathbf{B}$  and  $\mathbf{C}$ , where

$$a = \mathbf{B} : \mathbf{C} = \sum_i \sum_j \mathbf{B}_{ij} \mathbf{C}_{ij}.$$

The authors extend this notation to the form seen in (8),  $\mathbf{A} = \mathbf{B} : \mathbf{C}$  to indicate that

$$\mathbf{A}_{ij} = \mathbf{B}_{ij} : \mathbf{C}. \quad (9)$$

In other words,  $\mathbf{B}$  is some matrix of sub-matrices, such that  $\mathbf{B}_{ij}$  itself is a matrix. Each entry of  $\mathbf{A}$  is the Frobenius inner product of the corresponding sub-matrix in  $\mathbf{B}$  with  $\mathbf{C}$ . In the case of (8),  $\mathbf{A}$  and  $\mathbf{C}$  are  $3 \times 3$  matrices while  $\mathbf{B}$  is a  $3 \times 3$  matrix of  $3 \times 3$  sub-matrices.

Let's define  $\mathbf{u}^* \triangleq \mathbf{E}\mathbf{u}$ . Equation (6) can be rewritten into the form of (7) and (8)

$$\mathbf{u}^*(\mathbf{u}) = \sum_j \left( \mathbf{I} \frac{\delta_{ij}}{\Delta t} + \frac{\beta \Delta t}{m_i} \frac{\partial^2 \Phi^n}{\partial \hat{\mathbf{x}}_i \partial \hat{\mathbf{x}}_j} \right) \Delta t \mathbf{u}_j = \mathbf{u}_i + \frac{\beta \Delta t}{m_i} \sum_j \frac{\partial^2 \Phi^n}{\partial \hat{\mathbf{x}}_i \partial \hat{\mathbf{x}}_j} (\Delta t \mathbf{u}_j) = \mathbf{u}_i - \frac{\beta \Delta t}{m_i} \delta \mathbf{f}_i, \quad (10)$$

where  $\Delta t \mathbf{u}_j$  is equivalent to  $\delta \mathbf{u}_j$ . Let's define

$$\delta \mathbf{F}_{Ep} \triangleq \sum_j \Delta t \mathbf{u}_j (\nabla w_{jp})^T \mathbf{F}_{Ep}^n,$$

given this vector  $\mathbf{u}_j$ , such that

$$\mathbf{A}_p = \frac{\partial^2 \Psi}{\partial \mathbf{F}_{Ep} \partial \mathbf{F}_{Ep}} : \delta \mathbf{F}_{Ep}.$$

In the technical report, the authors show that

$$\frac{\partial^2 \Psi}{\partial \mathbf{F}_{Ep} \partial \mathbf{F}_{Ep}} : \delta \mathbf{F}_{Ep} = 2\mu \delta \mathbf{F}_{Ep} - 2\mu \delta \mathbf{R}_{Ep} + \lambda J_{Ep} \mathbf{F}_{Ep}^{-T} (J_{Ep} \mathbf{F}_{Ep}^{-T} : \delta \mathbf{F}_{Ep}) + \lambda (J_{Ep} - 1) \delta (J_{Ep} \mathbf{F}_{Ep}^{-T}), \quad (11)$$

where as usual,  $\mathbf{F}_{Ep} = \mathbf{R}_{Ep} \mathbf{S}_{Ep}$  by polar decomposition, and  $J_{Ep} = \det(\mathbf{F}_{Ep})$ .

The matrix  $J_{Ep} \mathbf{F}_{Ep}^{-T}$  is the cofactor matrix of  $\mathbf{F}_{Ep}$ , so its entries can be expressed as polynomials in the entries of  $\mathbf{F}_{Ep}$ . So the  $3 \times 3$  matrix of  $3 \times 3$  sub-matrices  $\frac{\partial}{\partial \mathbf{F}_{Ep}} (J_{Ep} \mathbf{F}_{Ep}^{-T})$  can be expressed in terms of the entries of  $\mathbf{F}_{Ep}$ , which allows us to compute

$$\delta (J_{Ep} \mathbf{F}_{Ep}^{-T}) = \frac{\partial}{\partial \mathbf{F}_{Ep}} (J_{Ep} \mathbf{F}_{Ep}^{-T}) : \delta \mathbf{F}_{Ep}.$$

We also have to compute  $\delta \mathbf{R}_{Ep}$ :

$$\begin{aligned} \delta \mathbf{F}_{Ep} &= \delta \mathbf{R}_{Ep} \mathbf{S}_{Ep} + \mathbf{R}_{Ep} \delta \mathbf{S}_{Ep} \\ \Rightarrow \mathbf{R}_{Ep}^T \delta \mathbf{F}_{Ep} - \delta \mathbf{F}_{Ep}^T \mathbf{R}_{Ep} &= (\mathbf{R}_{Ep}^T \delta \mathbf{R}_{Ep}) \mathbf{S}_{Ep} + \mathbf{S}_{Ep} (\mathbf{R}_{Ep}^T \delta \mathbf{R}_{Ep}). \end{aligned} \quad (12)$$

Equation (12) might not seem especially helpful, but we can take advantage of the fact that  $\mathbf{R}_{Ep}^T \delta \mathbf{R}_{Ep}$  is skew symmetric, so it only has three independent entries. In fact, the left-hand side of (12) is also skew-symmetric. Let's define

$$\mathbf{V} \triangleq \mathbf{R}_{Ep}^T \delta \mathbf{F}_{Ep} - \delta \mathbf{F}_{Ep}^T \mathbf{R}_{Ep} = \begin{pmatrix} 0 & d & e \\ -d & 0 & f \\ -e & -f & 0 \end{pmatrix} \quad \text{and} \quad \mathbf{U} \triangleq \mathbf{R}_{Ep}^T \delta \mathbf{R}_{Ep} = \begin{pmatrix} 0 & a & b \\ -a & 0 & c \\ -b & -c & 0 \end{pmatrix}.$$

We have the values of  $\mathbf{V}$  and  $\mathbf{S}_{Ep} = \{s_{ij}\}$ . The three independent values of  $\mathbf{U}$  can be computed directly by solving the  $3 \times 3$  system

$$\begin{pmatrix} s_{00} + s_{11} & s_{21} & -s_{02} \\ s_{12} & s_{00} + s_{22} & s_{01} \\ -s_{02} & s_{10} & s_{11} + s_{22} \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} d \\ e \\ f \end{pmatrix}$$

Then,  $\delta \mathbf{R}_{Ep} = \mathbf{R}_{Ep} (\mathbf{R}_{Ep}^T \delta \mathbf{R}_{Ep})$ .